

(19) 日本国特許庁(JP)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平 8 - 2 2 7 3 7 2

(43) 公開日 平成8年(1996)9月3日

(51) Int. Cl. ⁶	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 12/00	5 3 1	7623-5 B	G 0 6 F 12/00	5 3 1 R
	5 2 0	7623-5 B		5 2 0 J

審査請求 未請求 請求項の数 4

F D

(全 1 8 頁)

(21) 出願番号 特願平7-55114

(22) 出願日 平成7年(1995)2月20日

(71) 出願人 000001007

キャノン株式会社

東京都大田区下丸子3丁目30番2号

(72) 発明者 田中 憲明

東京都大田区下丸子3丁目30番2号 キャノ
ン株式会社内

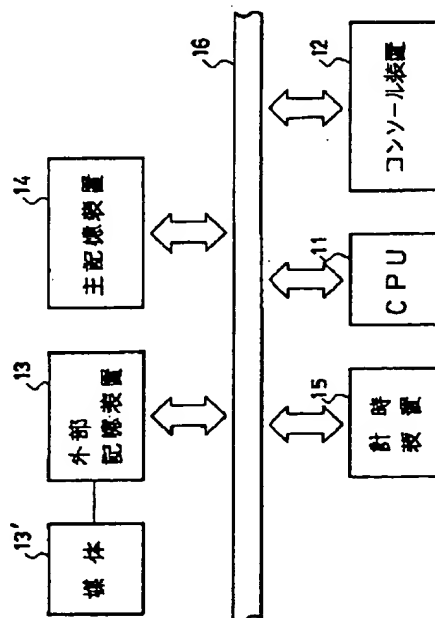
(74) 代理人 弁理士 川久保 新一

(54) 【発明の名称】 データ処理装置

(57) 【要約】

【目的】 何らかの原因によって失われたFATチェーンを回復し、ファイルを復元することができるデータ処理装置を提供することを目的とする。

【構成】 ファイルシステムの論理的整合性が正しくとれているかを検査する検査部101と、ブロック中のデータを論理的に解析する解析部105と、その結果をもとにブロックの使用順序を推定し、その結果をもとに正しいFATもしくはブロックの使用情報を記憶するファイル復元部102を有するように構成することによって、失われたFATチェーンを回復し、ファイルを復元するようにした。



【特許請求の範囲】

【請求項1】 記憶媒体に構築されたファイルシステム内のファイルを制御するデータ処理装置において、ファイルシステムの論理的整合性が正しいかどうかを検査する検査手段と、記憶媒体中のデータを論理的に解析する解析手段と、この解析結果に基づいて、あるファイルがどのブロックをどのような順序で利用しているかを示す情報を復元し、記憶媒体に書き込む順序管理情報復元手段を有することを特徴とするデータ処理装置。

【請求項2】 請求項1において、データファイルのレコード毎にファイル復元の手がかりとなる情報を記憶する記憶手段を有することを特徴とするデータ処理装置。

【請求項3】 請求項1または2において、上記検査手段と上記解析手段と上記順序管理情報復元手段の処理を、計時装置の出力に対応して実行する手段を有することを特徴とするデータ処理装置。

【請求項4】 請求項1または2において、上記検査手段と上記解析手段と上記順序管理情報復元手段の処理を、ファイルアクセス時に実行する手段を有する

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、磁気ディスク、光磁気ディスク等の記憶媒体上のファイルを操作するデータ処理装置において、ファイルシステムの修復機能をもつデータ処理装置に関するものである。

【0002】

【従来の技術】一般のデータ処理装置において、外部記憶等として利用される記憶媒体は、セクタまたはクラスタなどと呼ばれる一定の容量をもつ多数のブロックから構成されており、ファイルとして格納されたデータがどのブロックにどのような順序で格納されているかという情報を管理するため、媒体上にはファイルシステムが構築されている。

【0003】以下、ファイルシステムの例を図16～図18に基づいて説明する。

【0004】この例では、媒体上の記憶領域は、FAT、ディレクトリおよびデータ領域の3つに分割される。

【0005】そして、ディレクトリには、ファイルの名称、作成年月日などとともに、バイト数で表わされたファイルの大きさ、およびそのファイルのデータが格納されたブロックのうちの最初のものの番号が記憶されている。

【0006】また、ファイルとして格納されたデータが、どのブロックにどのような順番に格納されているかは、FAT（ファイル・アロケーション・テーブル）とよばれる領域に記憶されている。

【0007】このFATは、データの格納されるブロッ

クの1つ1つに対応するレコードを有している。そして、あるブロックに対応するFAT上のレコードには、当該ブロックに格納されたデータの次のデータが格納されるブロックの番号が記憶される。

【0008】また、データが格納された最後のブロックに対応するFATには、それが終端ブロックであることを示すコード（“EOC”と表わす）が記憶されている。

【0009】ディレクトリに記憶された先頭ブロックから、FATに“EOC”が書かれた終端ブロックまでのブロック番号の並びをFATチェーンという。

【0010】例えば、図17～図18において、ファイル“foo”は、ブロック番号25ではじまり、26、27、28、50、51の合計6の大きさ1024バイトのブロックに格納された5825バイトの大きさのファイルである。

【0011】

【発明が解決しようとする課題】ここで、誤操作またはハードウェア／ソフトウェアの欠陥、あるいは停電等によりファイルシステムの整合が失われると、データ処理装置はファイルへのアクセスを正常に行うことができなくなる。

【0012】例えば、図19に示すように、FATのデータが失われた時、ファイル“foo”のデータにアクセスしようとしても、ブロック番号25に対応するFATに正しい次ブロックの番号が記憶されていないと、データ処理装置は、続くデータの格納されたブロックの番号が分からないので、データを読み出すことができない。

【0013】このように、上記従来例では、ファイルシステムのFATに代表されるブロックの利用順序を管理する情報が失われた時には、ファイルとして記憶されたデータを復元することができなかった。

【0014】また、データ処理装置の動作中にファイルシステムの損傷が発生したとき、損傷したファイルシステムに対して、さらにデータの変更・書き込みを行うことによって、損傷を拡大してしまう危険が数多く存在していた。

【0015】本発明は、何らかの原因によって失われたFATチェーンを回復し、ファイルを復元することができるデータ処理装置を提供することを目的とする。

【0016】

【課題を解決するための手段】本発明は、ファイルシステムの論理的整合性が正しくとれているかを検査する検査手段と、ブロック中のデータを論理的に解析する解析手段と、その結果をもとにブロックの使用順序を推定し、その結果をもとに正しいFATもしくはブロックの使用情報を記憶するファイル復元手段を有するように構成することによって、失われたFATチェーンを回復し、ファイルを復元するものである。

【0017】また、上記検査手段、解析手段、および復元手段を、計時装置の出力に対して実行するように構成することにより、システムの動作中にファイルシステムに損傷が起こっても、即時に修復が行われ、システム全体の信頼性を向上するものである。

【0018】また、上記検査手段、解析手段、および復元手段を、ファイルアクセス時に実行するように構成することにより、システムの信頼性を向上するものである。

【0019】

【実施例】図1は、本発明の第1実施例におけるハードウェアの構成を示すブロック図であり、図2は、この第1実施例における機能要素を概念的に示すブロック図である。

【0020】図1において、CPU11は、本実施例装置の全体を制御するものであり、特に本実施例において、外部記憶装置13へのアクセス、演算等の制御を行うものである。コンソール装置12は、操作者とのインターフェイスをとるためのものであり、外部記憶装置13は、装置に固定または交換可能な記憶媒体13'に対して各種のデータを記憶し、読み出すものである。主記憶装置14は、CPU11によって実行されるコードおよび処理の対象となるデータを格納するものであり、時計装置15は、現在時刻を出力する計時装置である。

【0021】また、上記のCPU11、コンソール装置12、外部記憶装置13、主記憶装置14、時計装置15は、バス16によって接続されており、これを通じてデータや制御情報のやりとりが行われる。

【0022】次に、図2に示す各手段は、本データ処理装置が動作する場合に、CPU11によって実行される各コードにより構成されるものであり、これらは主記憶装置14上に格納されている。

【0023】まず、ファイルシステム検査手段101は、ファイルシステムの整合を検査するものであり、ファイルシステム復元手段102は、異常の発生したファイルシステムを修復するものである。また、FATアクセス103は、FAT、すなわち媒体13'上のデータの順序管理情報をアクセスするためのものであり、データブロックアクセス手段104は、媒体13'上のデータが格納されたブロックをアクセスするためのものである。

【0024】日時データ解析手段105は、データブロック中から日時を記憶した領域を抽出し、その内容を解析するものである。周期的実行制御手段106は、上記ファイルシステム検査手段101およびファイルシステム復元手段102の動作を時計装置15を参照しながら周期的に制御するものである。コンソール107は、ユーザからの入力を受け付ける手段であり、ファイルアクセス制御手段108は、ファイルアクセスを制御する手段である。

【0025】以上の各手段101～108の機能は、本実施例においては主記憶装置14上の命令コードをCPU11が解釈・実行することによって実現される。

【0026】命令コードは、通常は外部記憶装置13上の媒体に格納され、必要に応じて主記憶装置14中に読み込まれる。また、主記憶装置14中のROM領域に恒久的に記憶されている場合もある。

【0027】次に、本実施例におけるファイルシステムの構成は、上述の図16～図19で説明した従来例と同様であるので説明は省略する。

【0028】次に、上述のようなファイルシステムにおける障害の検出方法について説明する。図20は、本実施例における障害検出動作を示すフローチャートである。

【0029】ファイルシステム上に発生した障害は、ファイルシステム検査部101によって、以下のようにして検出される。

【0030】まず、CPU11は、ディレクトリに書かれているファイルの大きさと先頭ブロック番号を主記憶装置14上に読み出す(S1)。

【0031】続いて、CPU11は、FATアクセス部102を用いて、該当するブロックに対応するFATに書かれている値を読み出し、これを繰り返しながらFATのチェーンをたどっていく(S2～S5)。

【0032】そして、読み出された値が、その媒体におけるブロック番号として不適切な値であった場合(S3)、または、そのファイルあるいは別のファイルで既に使用されたブロック番号であった場合(S4)、または、ブロックの合計で格納できるデータのサイズがディレクトリに記憶されているファイルサイズと合わなかった場合(S6)、または、どのディレクトリからも参照されなかったFATチェーンが存在していた場合(S7)などには、CPU11は、ファイルシステムに異常があるものとして(S9)、以下に述べるファイルの復元動作を行う。

【0033】なお、以上のような障害検出の結果は、コンソール装置12、主記憶装置14、または外部記憶装置13に対して出力される。

【0034】次に、本実施例におけるファイルの復元動作について説明する。図3は、本実施例におけるファイル復元動作を示すフローチャートである。

【0035】まず、障害の発生が検出されると、ファイルシステム復元手段102による処理が実行される。

【0036】なお、ここでは、固定レコード長を有するファイルのFATのデータ、および、ディレクトリに記憶された先頭のブロック番号が失われた場合のファイルの復元について説明する。

【0037】図4は、データファイルのレコードの内容を示す説明図である。

【0038】レコードの長さは800バイトで、レコー

ド中の17バイト目から32バイト目には、レコードの記憶日時を図5に示すフォーマットで記憶する構成としている。そして、本実施例では、日付のデータが特定の位置に記憶されているブロックを探索し、この結果、探索されたブロックをもとにファイルの復元を実行するのである。

【0039】最初に、先頭ブロックの探索を行う。

【0040】CPU11は、データブロックアクセス手段104を用いて、媒体13'中のブロックのデータを全て読みながら(S11)、ブロックの先頭から16バ
10 イト目から日時データ解析手段105によって31バイト目の間に入っているデータを解析し、日時のデータとして矛盾のないものであるかどうかを判定する(S12)。

【0041】ここで、日時データ解析手段105による解析は以下のように行われる。

【0042】まず、図4に示すように、日時の5バイト目から2バイトにはASCIIキャラクタで月の値が記憶されている。月の値であるから、その数字は01から12の間になければならない。したがって、この値がも
20 しこの範囲の外にあれば、そのブロックは当該データファイルの先頭ブロックではないことがわかる。

【0043】また、日時の7バイト目から2バイトにはASCIIキャラクタで日の値が記憶されている。日の値であるから、その数字は01から31の間になければならない。したがって、この値がもしこの範囲の外にあれば、そのブロックは当該データファイルの先頭ブロックではないことがわかる。

【0044】以下、同様の操作を年・時・分・秒に対して実行する。そして、上記解析の結果、日時の記憶データとしてありうる数値が記憶されていれば、そのブロックが当該データファイルの先頭ブロックであると判断し、ディレクトリ上の先頭ブロックが記憶される位置に、そのブロックの番号を書き込む(S13)。

【0045】次に、ファイルの内容が記憶された2番目のブロックの探索を行う。

【0046】これは、例えば1つのブロックに収容されるデータの大きさが1024バイトであるような記憶媒体を考えると、レコードサイズ800バイトであるから、図5に示すように、2番目のブロックでは593バ
40 イト目に日付のデータが現れることがわかる。

【0047】そこで、すべてのブロックのなかから、593バイト目から16バイトに日付のデータが記憶されているブロックを、先頭ブロックの探索時と同様にして探索する(S12)。

【0048】そして、もし見つければ、そのブロックの番号をFATアクセス手段103を用いて、先頭ブロックに対応するFATに書き込む(S13)。

【0049】同様に3番目のブロックの探索は、369
50 バイト目に日付のデータのあるブロックを探索すること

によって行われる。

【0050】以下、同様の手続きを実行し、ブロックを見つけられなくなるか、ブロックに収容できるデータのバイト数がディレクトリに記憶されたファイルサイズに達したときは(S14)、最後のブロックに対応するFATに最終ブロックであることを示すコード“EOC”を書き込む(S15)。

【0051】以上のようにして、ファイルの復元が完了する。

【0052】次に、上述した検査処理および復元処理の起動について説明する。

【0053】ファイルシステム検査手段101およびファイルシステム復元手段102は、周期的実行制御手段106の出力に基づいて起動することができる。

【0054】周期的実行制御手段106は、計時装置15を参照し、操作者によって指定された時刻あるいは周期毎に、ファイルシステム検査手段101またはファイルシステム復元手段102を起動する。

【0055】また、上記検査・復元手段101、102は、操作者によるコンソール装置12の操作によって、指定されたタイミングで実行することもできる。

【0056】この場合は、指定されたタイミングでしか上記動作が行われなため、実行によるCPU時間やメモリが節約される。

【0057】また、上記検査・復元手段101、102は、システムによるファイルアクセス時、すなわち外部記憶装置13上のファイルの読み書きが行われる前後に実行することができる。

【0058】ファイルアクセス制御手段108は、ファイルへの書き込みやファイルからの読み出しなどに代表されるファイルアクセスを検知すると、ファイルシステム検査手段101とファイルシステム復元手段102を起動する。

【0059】この場合は、ファイルアクセスが実行されるたびにファイルシステムの検査・復元を行うことから、システム全体の信頼性を著しく高めることができる効果がある。

【0060】次に、本発明の第2実施例について説明する。

【0061】図6は、本発明の第2実施例におけるハードウェアの構成を示すブロック図であり、図7は、この第2実施例における機能要素を概念的に示すブロック図である。

【0062】図6において、CPU51は、本実施例装置の全体を制御するものであり、特に本実施例において、外部記憶装置53へのアクセス、演算等の制御を行うものである。コンソール装置52は、操作者とのインターフェイスをとるためのものであり、外部記憶装置53は、装置に固定または交換可能な記憶媒体53'に対して各種のデータを記憶し、読み出すものである。主記

憶装置 54 は、CPU 51 によって実行されるコードおよび処理の対象となるデータを格納するものであり、時計装置 55 は、現在時刻を出力する時計装置である。

【0063】また、上記の CPU 51、コンソール装置 52、外部記憶装置 53、主記憶装置 54、時計装置 55 は、バス 56 によって接続されており、これを通じてデータや制御情報のやりとりが行われる。

【0064】次に、図 7 に示す各手段は、本データ処理装置が動作する場合に、CPU 51 によって実行される各コードにより構成されるものであり、これらは主記憶装置 54 上に格納されている。

【0065】ファイルシステム検査手段 501、ファイルシステム復元手段 502、FAT アクセス手段 503、データブロックアクセス手段 504、コンソール 507、ファイルアクセス制御手段 508 は、上記第 1 実施例（図 2）のファイルシステム検査手段 101、ファイルシステム復元手段 102、FAT アクセス手段 103、データブロックアクセス手段 104、コンソール 107、ファイルアクセス制御手段 108 と同様である。

【0066】次に、レコード ID 番号解析手段 505 は、ファイルの復元時に記憶媒体上のブロック内のレコード ID 番号を抽出・解析するものであり、レコード番号作成手段 509 は、ファイルの書き込み時にレコード毎につけられるレコード ID 番号を作成するものである。なお、レコード ID 番号とは、レコードの先頭位置とその記憶番号を識別する ID のことである。

【0067】以上のように、この第 2 実施例の機能要素は、上記第 1 実施例の日時データ解析手段 105 の代わりに、レコード ID 番号解析手段 505 およびレコード番号作成手段 509 を有するものである。

【0068】また、各手段 501～509 の機能は、本実施例においては主記憶装置 54 上の命令コードを CPU 51 が解釈・実行することによって実現される。

【0069】命令コードは、通常は外部記憶装置上の媒体に格納され、必要に応じて主記憶装置中に読み込まれる。また、主記憶中の ROM 領域に恒久的に記憶されている場合もある。

【0070】次に、本実施例におけるファイルシステムの構成は、上記第 1 実施例と同様に上述の図 16～図 19 で説明した従来例と同様であるので説明は省略する。

【0071】また、本実施例における障害の検査、および検査・復元の起動動作についても、上記第 1 実施例と同様であるので説明は省略する。

【0072】次に、本実施例におけるファイルの復元動作について説明する。図 8 は、本実施例におけるファイル復元動作を示すフローチャートである。

【0073】まず、障害の発生が検出されると、ファイルシステム復元手段 502 による処理が実行される。

【0074】ここでは、レコード ID 番号をレコードの先頭に記憶したデータファイルの、FAT のデータおよ

びディレクトリに記憶された先頭のブロック番号が失われた場合の、ファイルの復元について説明する。

【0075】図 9 は、データファイルのレコードの内容を示す説明図である。

【0076】本実施例では、ファイルヘデータを書き込む際に、レコードの先頭にレコード ID 作成手段 509 によって作成されたレコード ID 番号がつけられて、媒体上に記憶される構成になっている。

【0077】また、レコード ID 番号のフォーマットは“RecIDxxx”（xxx にはレコード番号が入る）という形式になっている。

【0078】そして、本実施例では、このレコード ID 番号の記憶されたブロックを探索し、それをもとにファイルの復元を実行する。

【0079】まず、先頭ブロックの探索動作について説明する。

【0080】CPU 51 は、データブロックアクセス手段 504 を用いて、媒体 53' 上のブロックを 1 つずつ読み取る（S21）。続いて、レコード ID 解析部 505 を用いて、ブロック中に“RecIDxxx”のフォーマットで最初のレコードであることを示す“RecID001”という文字列が先頭に記憶されたブロックを探索する（S22）。

【0081】そして、もし見つければ、それが先頭ブロックであると判断して、ディレクトリ上の先頭ブロックが記憶される位置に、そのブロックの番号を書き込む（S23）。

【0082】次に、2 番目のブロックの探索を行う。これは、レコード ID 解析手段 505 を用いて、先頭ブロック中の“RecIDxxx”フォーマットの文字列を全て検索し、最後のものが“RecID003”であれば、次に“RecID004”という文字列を含むブロックを全媒体上から探し出す（S21、S22）。

【0083】そして、もし見つければ、そのブロックの番号を FAT アクセス手段 503 を用いて、先頭ブロックに対応する FAT に書き込む（S23）。

【0084】また、3 番目以降のブロックの探索も同様である。そして、ブロックを見つけられなくなるか、ブロックに収容できるデータのバイト数がディレクトリに記憶されたファイルサイズに達したときは（S24）、最後のブロックに対応する FAT に最終ブロックであることを示すコード“EOC”を書き込む。

【0085】以上のようにして、ファイルの復元が完了する。

【0086】以上のように、この第 2 実施例では、レコード境界とブロックの境界が一致した場合でも、レコード ID に記憶された情報をもとに、FAT チェインを復元することができる。

【0087】次に、本発明の第 3 実施例について説明する。

【0088】図11は、本発明の第3実施例におけるハードウェアの構成を示すブロック図であり、図12は、この第3実施例における機能要素を概念的に示すブロック図である。

【0089】図11において、CPU91は、本実施例装置の全体を制御するものであり、特に本実施例において、外部記憶装置93へのアクセス、演算等の制御を行うものである。コンソール装置92は、操作者とのインターフェイスをとるためのものであり、外部記憶装置93は、装置に固定または交換可能な記憶媒体93'に対して各種のデータを記憶し、読み出すものである。主記憶装置94は、CPU91によって実行されるコードおよび処理の対象となるデータを格納するものであり、計時装置95は、現在時刻を出力する計時装置である。画像データ伸長部97は、MH・MR・MMR等の公知の画像圧縮手段によって圧縮された画像データを伸長する。また、上記のCPU91、コンソール装置92、外部記憶装置93、主記憶装置94、計時装置95、画像データ伸長部97は、バス96によって接続されており、これを通じてデータや制御情報のやりとりが行われる。

【0090】次に、図12に示す各手段は、本データ処理装置が動作する場合に、CPU91によって実行される各コードにより構成されるものであり、これらは主記憶装置94上に格納されている。

【0091】ファイルシステム検査手段901、ファイルシステム復元手段902、FATアクセス手段903、データブロックアクセス手段904、コンソール907、ファイルアクセス制御手段908は、上記第1実施例(図2)のファイルシステム検査手段101、ファイルシステム復元手段102、FATアクセス手段103、データブロックアクセス手段104、コンソール107、ファイルアクセス制御手段108と同様である。

【0092】また、伸長画像判定解析手段905は、ファイルの復元時に記憶媒体上のブロック内の画像データが画像データ伸長部97によって正しく伸長されたかどうかを判定するものである。

【0093】以上のように、この第3実施例の機能要素は、上記第1実施例の日時データ解析手段105の代わりに、伸長画像判定解析手段905を有するものである。

【0094】また、各手段901～908の機能は、本実施例においては主記憶装置94上の命令コードをCPU91が解釈・実行することによって実現される。

【0095】命令コードは、通常は外部記憶装置上の媒体に格納され、必要に応じて主記憶装置中に読み込まれる。また、主記憶装置中のROM領域に恒久的に記憶されている場合もある。

【0096】次に、本実施例におけるファイルシステムの構成は、上記第1実施例と同様に上述の図16～図19で説明した従来例と同様であるので説明は省略する。

【0097】また、本実施例における障害の検査、および検査・復元の起動動作についても、上記第1実施例と同様であるので説明は省略する。

【0098】次に、本実施例における圧縮画像ファイルの復元動作について説明する。

【0099】まず、障害の発生が検出されると、ファイルシステム復元手段902の処理が実行される。図13は、本実施例におけるファイル復元動作を示すフローチャートである。

【0100】なお、ここでは、MH・MR・MMRなどの公知の画像圧縮手段によって圧縮された画像データファイルのFATのデータが失われた場合のファイルの復元について説明する。

【0101】上述のように、ディレクトリには、データが格納された先頭のブロックの番号が書かれているが、1つのブロックに収容できない大きさのファイルの場合、すなわちファイルの内容が複数のブロックにわたって格納されている場合には、FATのデータが失われると、2つ目以降のブロックの番号がわからなくなってしまう。

【0102】そこで、まず先頭ブロックに続く、2番目のブロックを探索しなければならない。

【0103】図14に示すように、圧縮画像データファイルの先頭には、画像の圧縮方法などとともに、伸長後の画像の縦・横のピクセル数などが記述されたヘッダが記憶されており、続いて実際の圧縮されたデータが書き込まれている。

【0104】まず、CPU91は、画像データ伸長部97を用いて、先頭ブロック内のヘッダに続く領域に記憶されている圧縮画像データを1ラインずつブロックの終端まで伸長する。

【0105】図15に示すように、ブロックの終端では一般にラインの伸長の途中で終了することが多いと予想される。

【0106】そこで、媒体中のブロックのデータを1つずつ読み込み(S31)、実際の伸長動作を繰り返しながら、中断したラインに続けてデータの伸長が可能になるようなデータが記憶されているブロックを探す(S32)。

【0107】このとき、ヘッダに記憶された横方向のピクセル数を、ラインの伸長結果として得られた横ピクセル数と比較し、それが一致するかどうかを調べること、そのブロックが目的のものであるかどうかを判定することができる。

【0108】そして、伸長画像判定手段905によって、上記の方法で見つけられたブロックの番号が、FATアクセス手段903を用いて、先頭ブロックに対応するFATに書き込まれる(S33)。

【0109】以上のブロックの探索を、全伸長ライン数が縦方向のピクセル数に達するまで繰り返す(S3

4)。

【0110】そして、最後のブロックに対応するFATには、終端ブロックであることを示すコード“EOC”が書き込まれる(S35)。

【0111】以上のようにして、圧縮画像データファイルの復元が完了する。

【0112】

【発明の効果】以上説明したように、本発明によれば、ブロック中のデータを論理的に解析し、その結果をもとにデータの格納されたブロックを搜索するように構成することにより、FATが失われるような深刻なファイルシステム上の障害が発生した場合にも、データを失うことなくファイルの復元を行うことが可能になる。

【0113】また、媒体の検査・解析および復元動作を計時装置の出力に対応して動作するように構成することにより、システム全体の信頼性を著しく向上せしめることができる。

【0114】また、媒体の検査・解析および復元動作をファイルアクセス時に動作するように構成することにより、システム全体の信頼性を著しく向上せしめることができる。

【図面の簡単な説明】

【図1】本発明の第1実施例におけるハードウェアの構成を示すブロック図である。

【図2】上記第1実施例における機能要素を概念的に示すブロック図である。

【図3】上記第1実施例におけるファイル復元動作を示すフローチャートである。

【図4】上記第1実施例におけるファイルのレコードの内容を示す説明図である。

【図5】上記第1実施例における日付データの位置を示す説明図である。

【図6】本発明の第2実施例におけるハードウェアの構成を示すブロック図である。

【図7】上記第2実施例における機能要素を概念的に示

すブロック図である。

【図8】上記第2実施例におけるファイル復元動作を示すフローチャートである。

【図9】上記第2実施例におけるファイルのレコードの内容を示す説明図である。

【図10】上記第2実施例におけるブロックとレコードID番号を示す説明図である。

【図11】本発明の第3実施例におけるハードウェアの構成を示すブロック図である。

【図12】上記第3実施例における機能要素を概念的に示すブロック図である。

【図13】上記第3実施例におけるファイル復元動作を示すフローチャートである。

【図14】上記第3実施例における圧縮画像データファイルを示す説明図である。

【図15】上記第3実施例のブロック境界における画像データを示す説明図である。

【図16】ファイルシステムにおける媒体上の領域分割を示す説明図である。

【図17】ファイルシステムにおけるディレクトリを示す説明図である。

【図18】ファイルシステムにおけるFATとデータを示す説明図である。

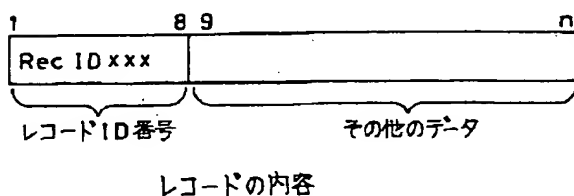
【図19】ファイルシステムにおいて、FATが失われた場合のデータの状態を示す説明図である。

【図20】上記各実施例におけるファイルシステムの検査動作を示すフローチャートである。

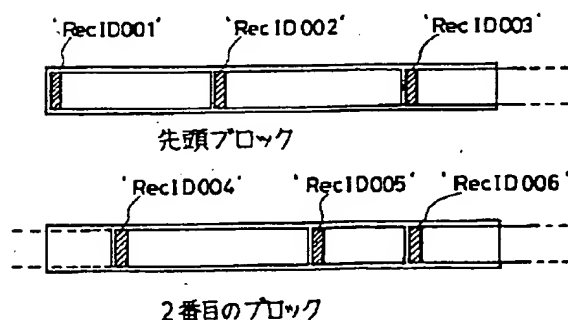
【符号の説明】

- 11…CPU、
- 12…コンソール装置、
- 13…外部記憶装置、
- 13'…記憶媒体、
- 14…主記憶装置、
- 15…計時装置、
- 16…バス。

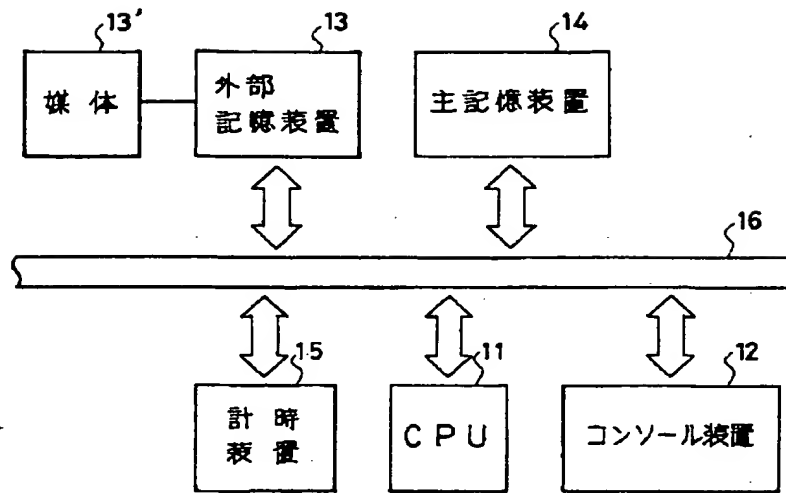
【図9】



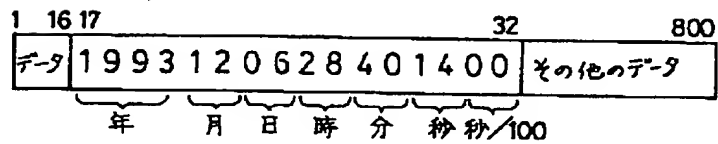
【図10】



【図1】

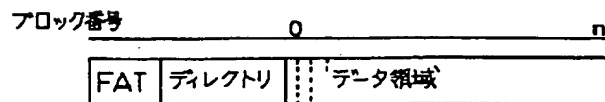


【図4】



レコードの内容

【図16】



媒体上の領域分割

【図17】

foo	93-12-24	---	5825	25
bar	93-12-25	---	7502	44

ファイル名

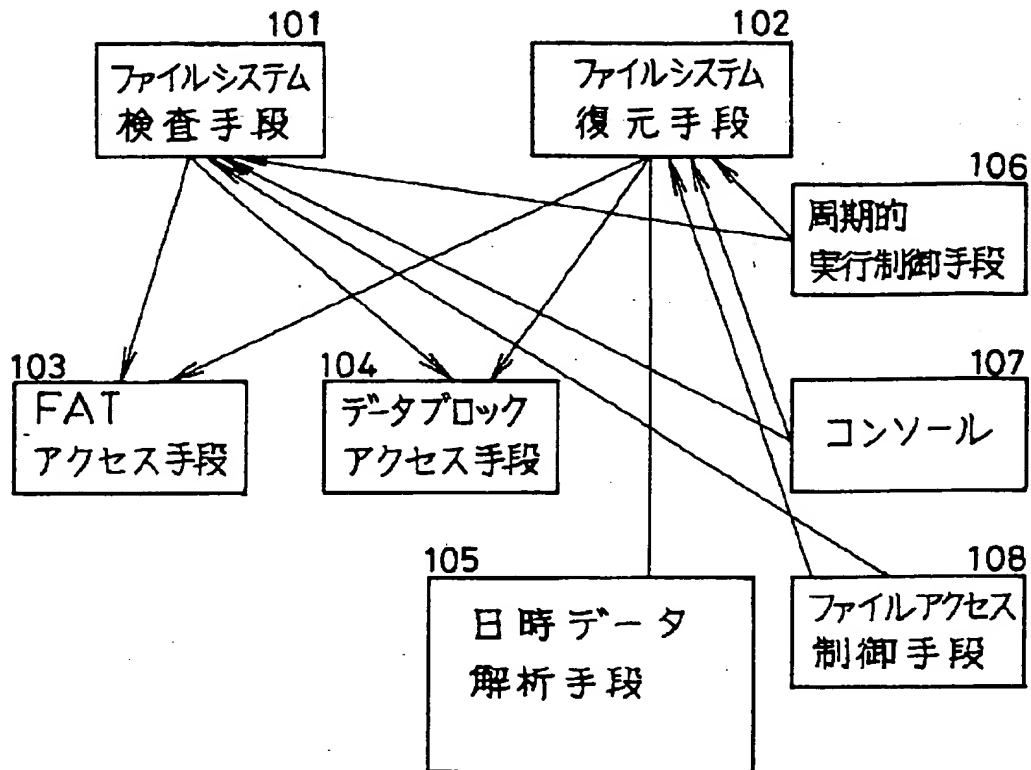
日付

ファイルサイズ

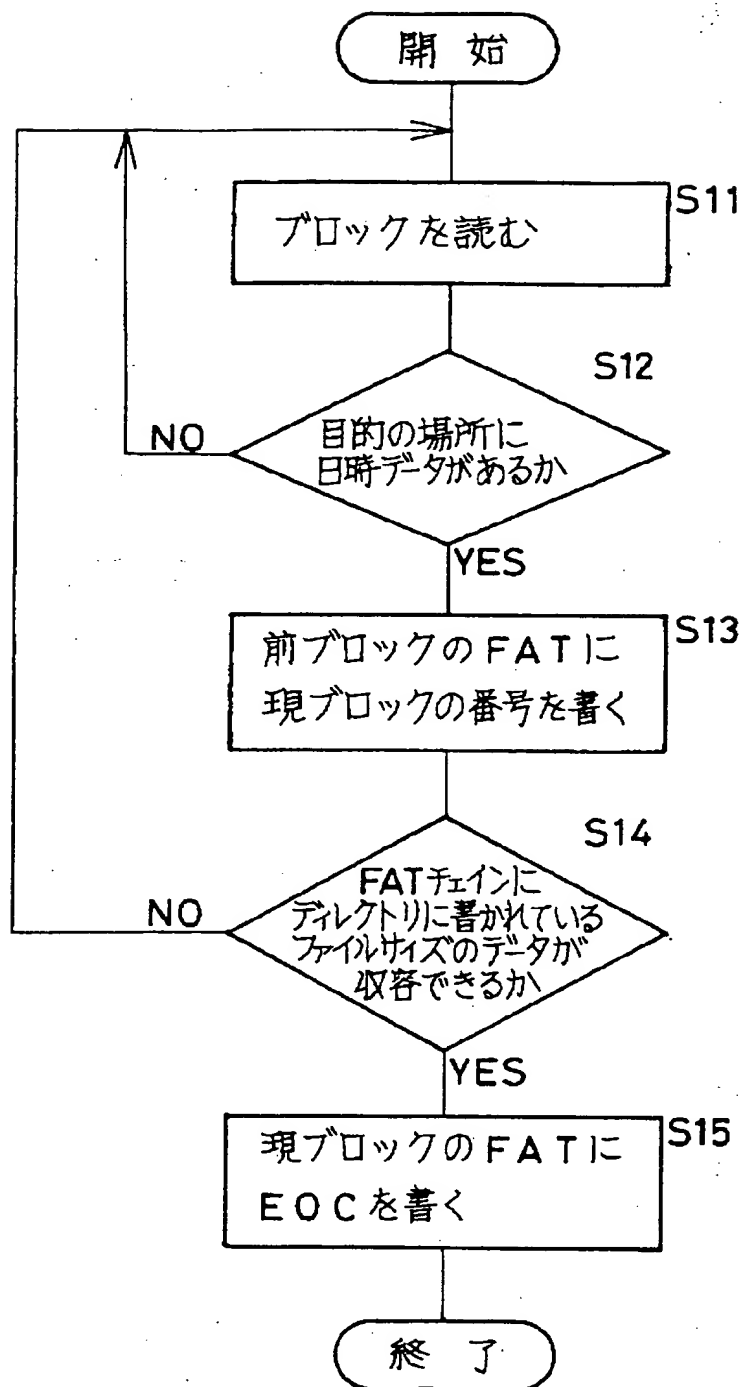
先頭ブロック
番号

ディレクトリ

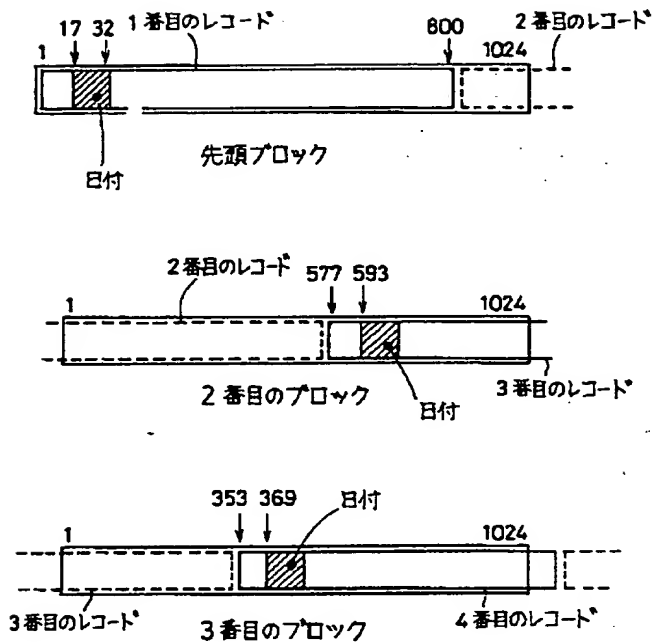
【図 2】



【図 3】

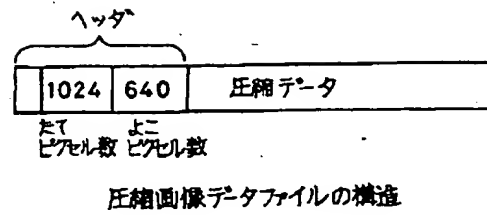


【図 5】



ブロック中の日付データの位置

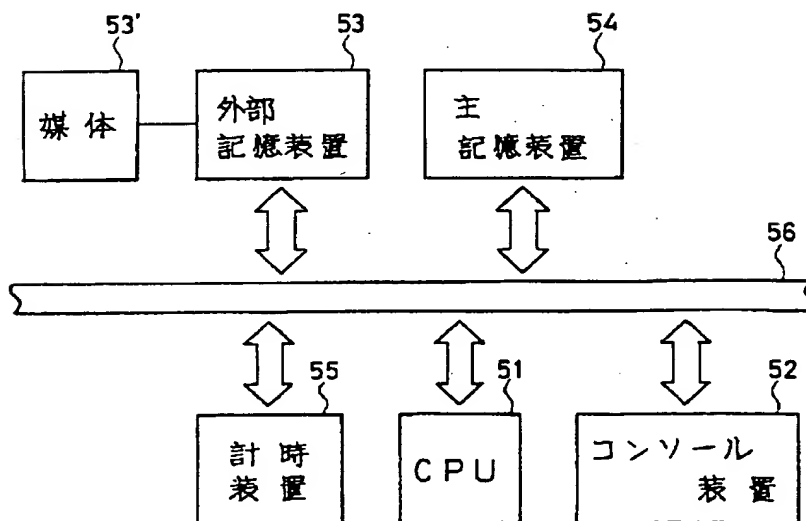
【図 14】



K3167

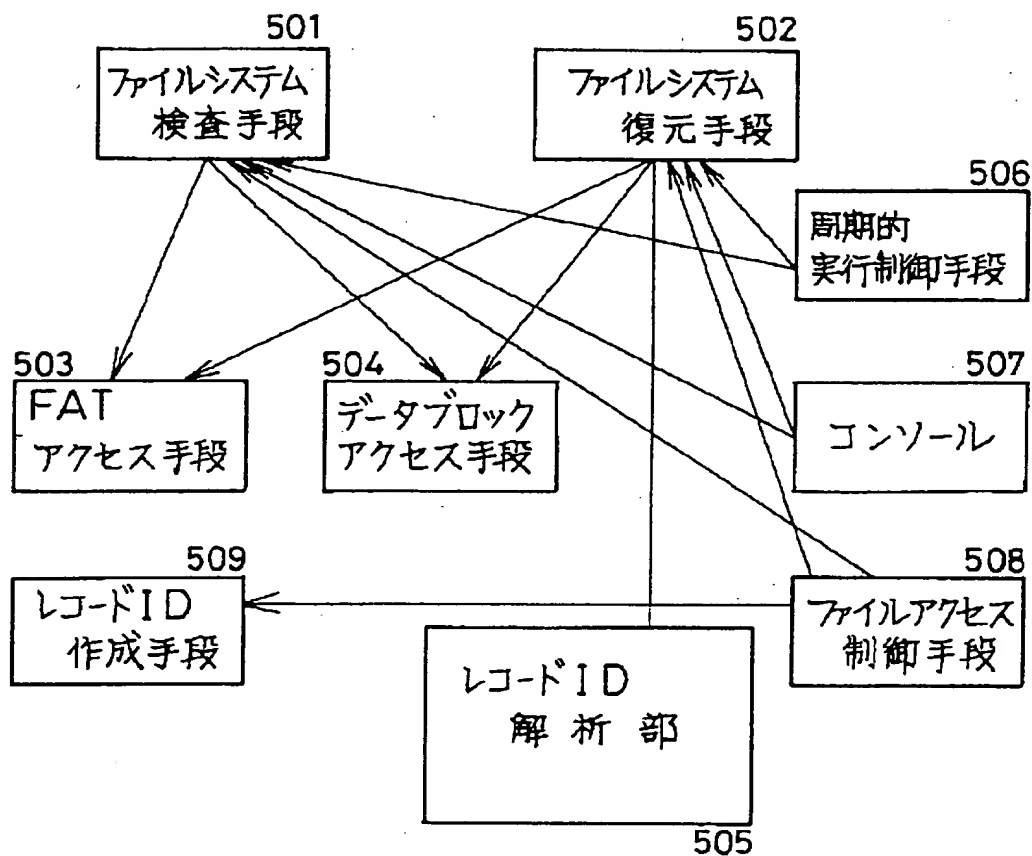
K3167

【図 6】

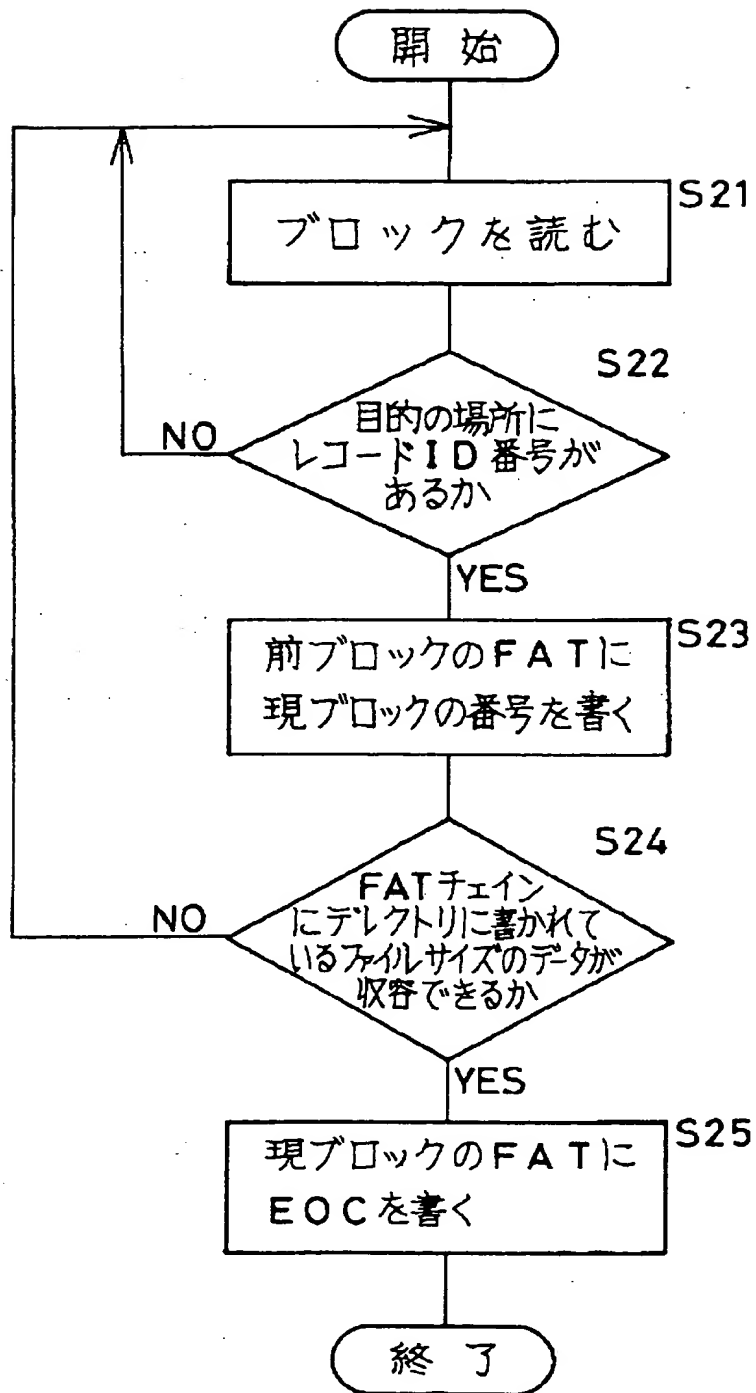


K3167

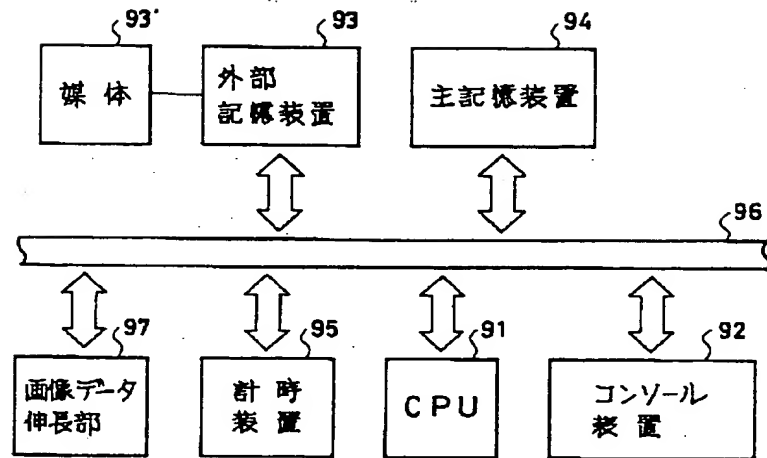
【図 7】



【図 8】

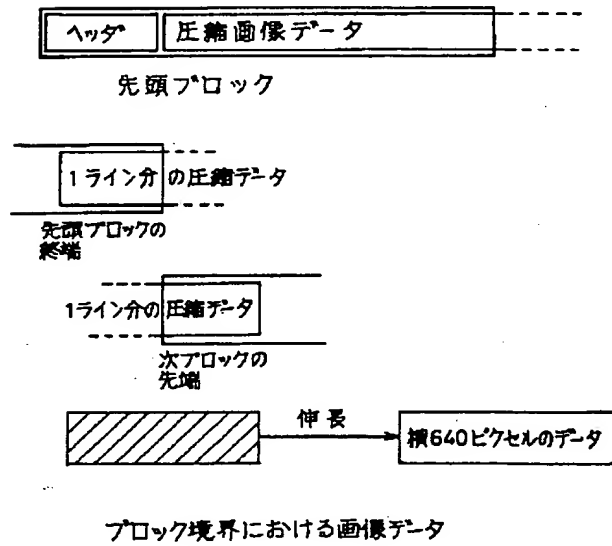


【図11】



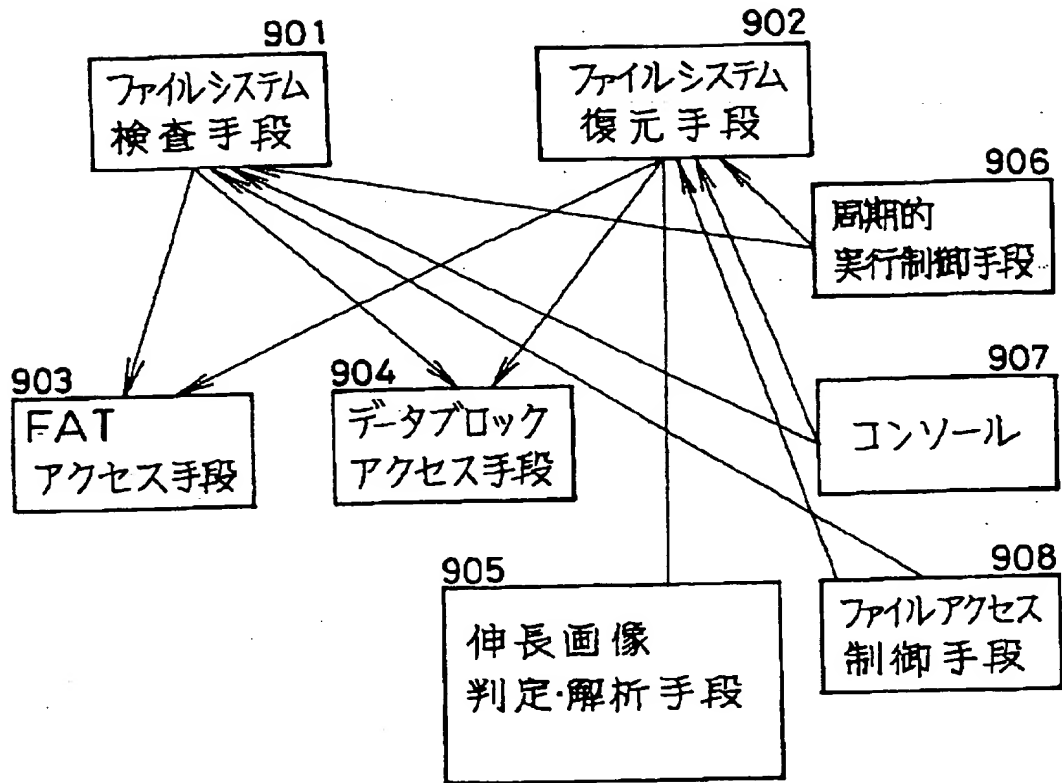
K3167

【図15】

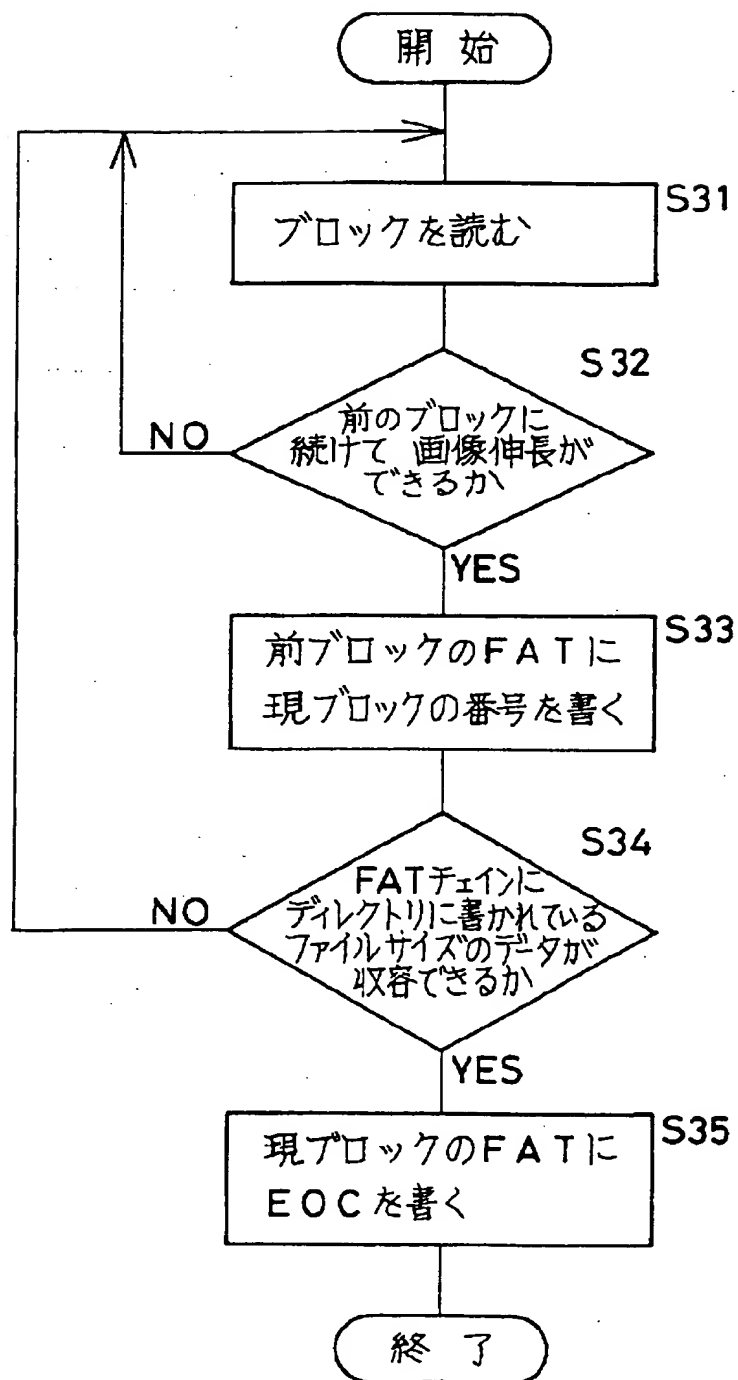


K3167

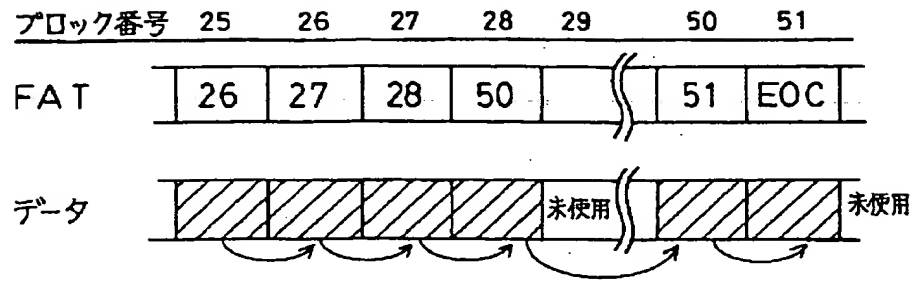
【図12】



【図 13】



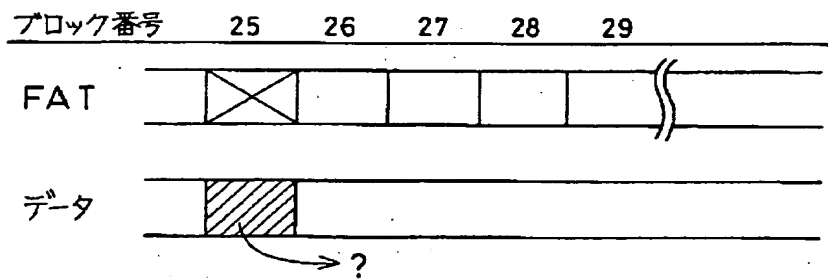
【図 18】



FATとデータ

K3167

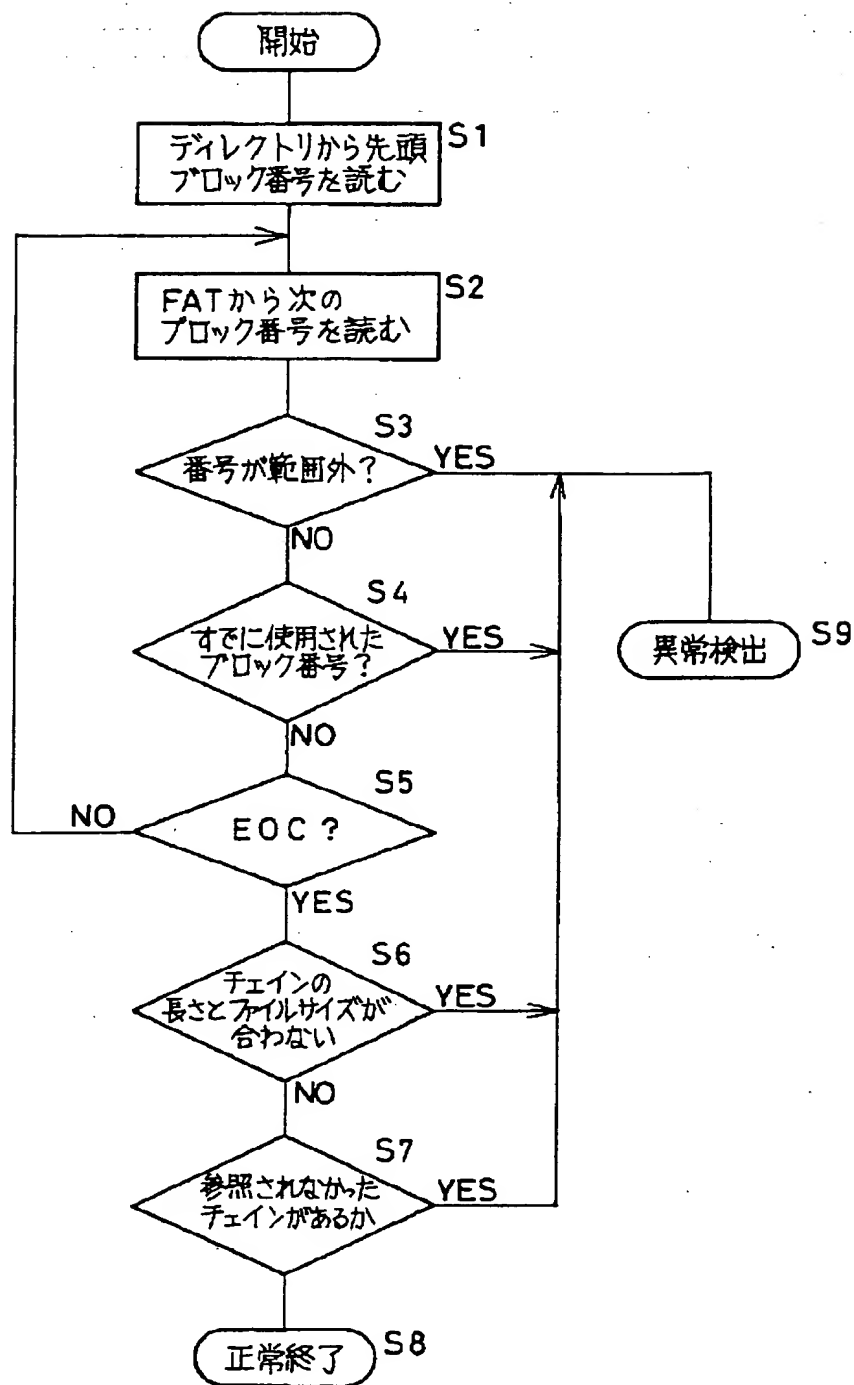
【図 19】



FAT が失われた場合

K3167

【図 2 0】



ファイルシステム検査

K3167

Title of the Prior Art

Japanese Published Patent Application No. Hei.8-227372

Date of Publication: September 3, 1996

Inventor: N. Tanaka

Concise Statement of Relevancy

Translation of a first embodiment

[0019]

[Embodiment]

Figure 1 is a block diagram illustrating the structure of hardware according to the first embodiment of the present invention. Figure 2 is a block diagram conceptually illustrating functional elements according to the first embodiment of the present invention.

[0020]

In figure 1, a CPU 11 controls the whole apparatus of this embodiment, and especially in this embodiment, it accesses to an outer storage apparatus 13, and controls an operation or the like. A console apparatus 12 is for interfacing with an operator, and the outer storage apparatus 13 stores each kind of data in a storage medium 13' which can fix on the apparatus or be exchanged, and reads out the same. A main storage apparatus 14 contains the code to be executed by the CPU 11 and data to be processed, and a clock apparatus 15 is a timing apparatus which outputs the present time.

[0021]

Further, the above CPU 11, console apparatus 12, outer storage apparatus 13, main storage apparatus 14, and clock apparatus 15 are

THIS PAGE BLANK (USPTO)

connected by a bus 16, and data and control information are exchanged therethrough.

[0022]

Next, each means shown in figure 2 is constituted of each code to be executed by the CPU 11 when the data processing apparatus operates, and these are stored on the main storage apparatus 14.

[0023]

Initially, a file system check means 101 checks matching of a file system, and a file system restoration means 102 recovers a file system where abnormalities occurred. Further, an FAT access 103 is the one for accessing the FAT, that is, order management information of the data on the medium 13', and a data block access means 104 is the one for accessing a block where the data on the medium 13' is contained.

[0024]

A date and hour data analyzing means 105 extracts the data on an area, where the date and hour are stored, from the data block, and analyzes its contents. A periodic execution control means 106 periodically controls the operation of the file system check means 101 and the file system restoration means 102 while referring to the timing apparatus 15. A console 107 is a means for accepting an input from a user, and a file access control means 108 is a means for controlling a file access.

[0025]

The function of the each above-described means 101-108 is

THIS PAGE BLANK (USPTO)

realized when the CPU 11 interprets and executes an order code on the main storage apparatus 14, in this embodiment.

[0026]

The order code is usually contained in the medium on the outer storage apparatus 13, and is read in the main storage apparatus 14 as needed. Further, there are some cases when the order code is permanently stored in a ROM area in the main storage apparatus 14.

[0027]

Next, as a structure of the file system in this embodiment is the same as conventional examples as explained in the above-mentioned figures 16 to 19, the description is omitted.

[0028]

Next, a method for detecting failures in the file system as above-mentioned will be described. Figure 20 is a flowchart illustrating a failure detection operation in this embodiment.

[0029]

A failure which occurs on the file system is detected by the file system check unit 101 as follows.

[0030]

Initially, the CPU 11 reads out the size of the file written in a directory and the head block number on the main storage apparatus 14 (S1).

[0031]

Successively, the CPU 11 reads out the value written in the FAT corresponding to the appropriate block; and while repeating this,

THIS PAGE BLANK (USPTO)

traces a chain of the FAT (S2-S5).

[0032]

Thus, when the read value is an inappropriate value as the block number in the medium (S3), when it is the block number already used for the file or the other file (S4), when the size of data which can be contained by using the total block number does not match the file size stored in the directory (S6), or when the FAT chain which is not seen by any directory exists (S7), the CPU 11 regards the file system as having abnormalities (S9) and then performs a restoring operation of the file as follows.

[0033]

Further, the result of the failure detection as above-mentioned is output to the console apparatus 12, the main storage apparatus 14, or outer storage apparatus 13.

[0034]

Next, the restoring operation of the file in this embodiment will be described. Figure 3 is a flowchart illustrating the file restoring operation in this embodiment.

[0035]

Initially, when the occurrence of the failure is detected, processing is executed by the file system restoration means 102.

[0036]

Further, a description will be given of the FAT data of the file having the fixed record length, and the file restoration when the head block number stored in the directory is lost, hereinafter.

THIS PAGE BLANK (USPTO)

[0037]

Figure 4 is an explanatory diagram illustrating the contents of a record of the data file.

[0038]

The record length is 800 bytes, and the stored date and hour of the record are stored in 17th to 32nd bytes in the record by a format shown in figure 5. Thus, in this embodiment, the block where the data of the date is stored in a specific position is searched. Consequently, the file restoration is executed on the basis of the searched block.

[0039]

Initially, the head block is searched.

[0040]

While the CPU 11 reads all the data of the block in the medium 13' by using the data block access means 104 (S11), it analyzes the data inserted in the one from the 16th byte from the head of the block to the 31st byte by the date and hour data analyzing means 105, and determines whether the data is consistent as the data of the date and hour (S12).

[0041]

Here, an analysis by the date and hour data analyzing means 105 is performed as follows.

[0042]

Initially, as shown in figure 4, the month value is stored in 2 bytes from the 5th byte of the date and hour by an ASCII character.

THIS PAGE BLANK (USPTO)

The number must be from 01 to 12 because it is the month value. Therefore, if this value is out of this range, it is found that the block is not the head block of the appropriate data file.

[0043]

Furthermore, the date value is stored in 2 bytes from the 7th byte of date and hour by the ASCII character. The number must be from 01 to 31 because it is the date value. Therefore, if this value is out of this range, it is found that the block is not the head block of the appropriate data file.

[0044]

Hereinafter, a similar operation is executed to year/hour/minute/second. Thus, as a result of the above analysis, a numerical value which is possible as the storage data of date and hour is stored, it is judged that the block is the head block of the appropriate data file, and the number of the block is written in the position where the head block is stored on the directory (S13).

[0045]

Next, the second block where the contents of the file are stored is searched.

[0046]

In consideration of the storage medium, for example, where the size of the data accommodated in one block is 1024 bytes, it is found that, as shown in figure 5, the data of the date appears in the 593rd byte in the second block, because the record size is 800 bytes.

[0047]

THIS PAGE BLANK (USPTO)

The block where the data of the date is stored in 16 bytes from the 593rd byte is searched among all the blocks, similarly as in searching the head block (S12).

[0048]

Thus, if it is found, the block number is written in the FAT corresponding to the head block by using the FAT access means 103 (S13).

[0049]

Similarly, the search for the 3rd block is performed in a manner of searching the block having the data of the date in the 369th byte.

[0050]

Hereinafter, similar steps are executed, and when the block is not found, or when the byte number of the data which can be accommodated in the block reaches the file size stored in the directory (S14), the code "EOC" indicating that the block is the final one is written in the FAT corresponding to the last block (S15).

[0051]

As above-described, the file restoration is completed.

[0052]

Next, the activation of the above-described checking process and restoring process will be described.

[0053]

The file system check means 101 and the file system restoration means 102 can be activated on the basis of an output from the periodic execution control means 106.

THIS PAGE BLANK (USPTO)

[0054]

The periodic execution control means 106 refers to the timing apparatus 15, and activates the file system check means 101 and the file system restoration means 102 at each time or period designated by the operator.

[0055]

Furthermore, the above check and restoration means 101 and 102 can be executed in the designated timing according to the operation of the console apparatus 12 by the operator.

[0056]

In this case, the CPU time and memory according to the execution are saved, because the above-described operation is performed only in the designated timing.

[0057]

Furthermore, when the above checking and restoring means 101 and 102 perform file access by using the system, that is, before or after reading and writing in the file on the outer storage apparatus 13, these means are executed.

[0058]

When the file access control means 108 detects file access represented by writing into the file, reading from the file or the like, the file system check means 101 and the file system restoration means 102 are activated.

[0059]

In this case, each time the file access is executed, the check

THIS PAGE BLANK (USPTO)

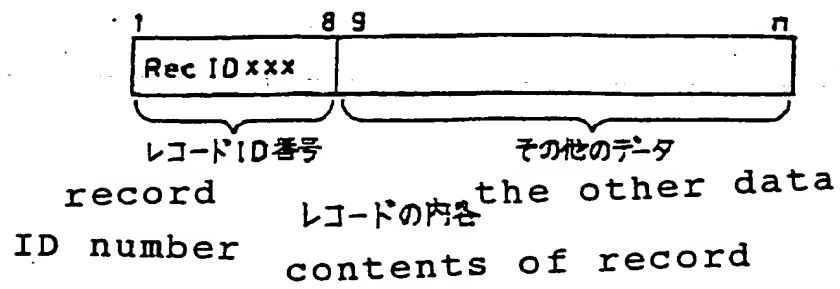
and the restoration of the file system are performed. Therefore, reliability in the whole system can remarkably increase.

Explanation

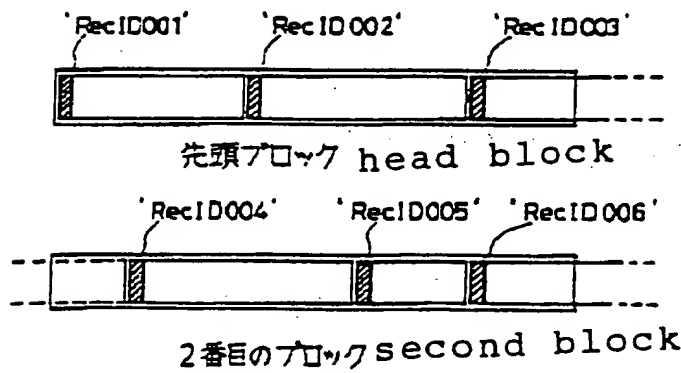
In a second embodiment, a record ID number is entered to the head of the record as shown in figure 9, and a description is given of a method for performing recovery processing by using the record having this record ID number, instead of the stored date and hour in the first embodiment.

THIS PAGE BLANK (USPTO)

【図9】 [Figure 9]



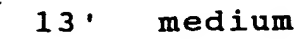
【図10】 [Figure 10]



ブロックとレコードID番号
block and record ID number

THIS PAGE BLANK (USPTO)

13 outer storage
apparatus



14 main storage
apparatus

15 timing apparatus

12 console apparatus

1 16 17 32 800
 データ 1.993 12 06 28 40 14 00 その他のデータ
 data 年 月 日 時 分 秒 秒/100
 ① ② ③ ④ ⑤ ⑥ ⑥
 レコードの内容
 contents of record

① year ② month
 ③ date ④ hour
 ⑤ minute ⑥ second

【図17】[Figure 17]

block number

ブロック番号

data region



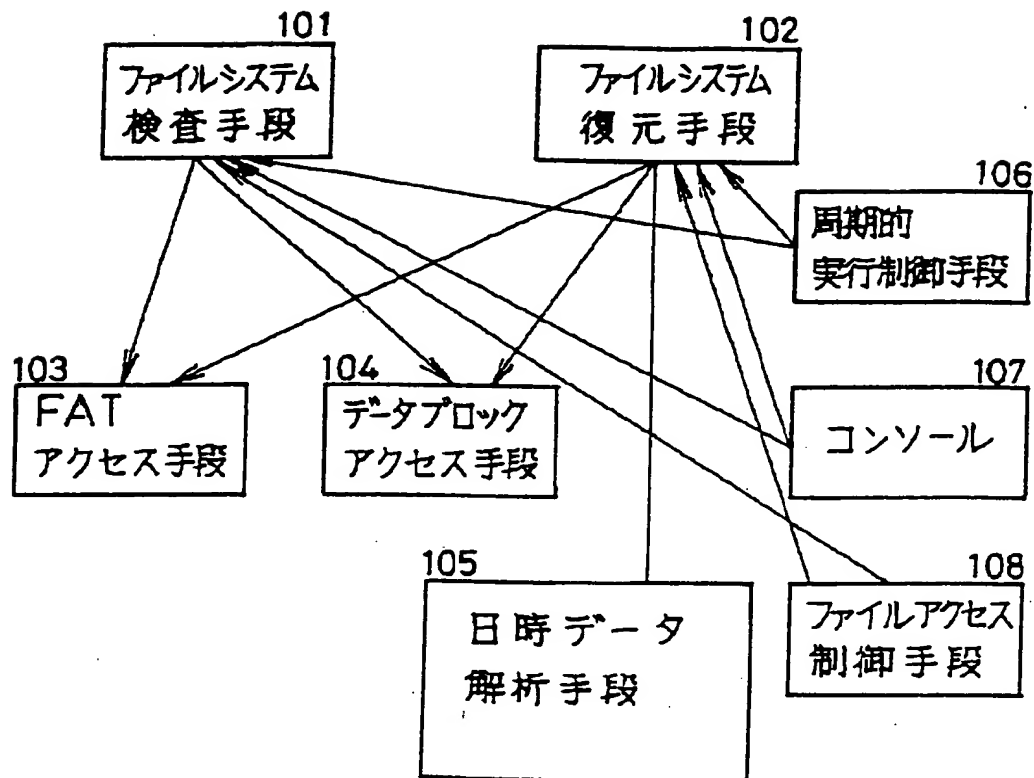
媒体上の領域分割

region division on medium

m	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;"> ファイル名 file name </div> <div style="text-align: center;"> 日付 date </div> </div>		<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;"> ファイルサイズ file size </div> <div style="text-align: center;"> 先頭ブロック 番号 head block number </div> </div>	
	ディレクトリ directory			

THIS PAGE BLANK (USPTO)

【図2】[Figure 2]



101 file system check means

102 file system restoration means

103 FAT access means

104 data block access means

105 date and hour data analyzing means

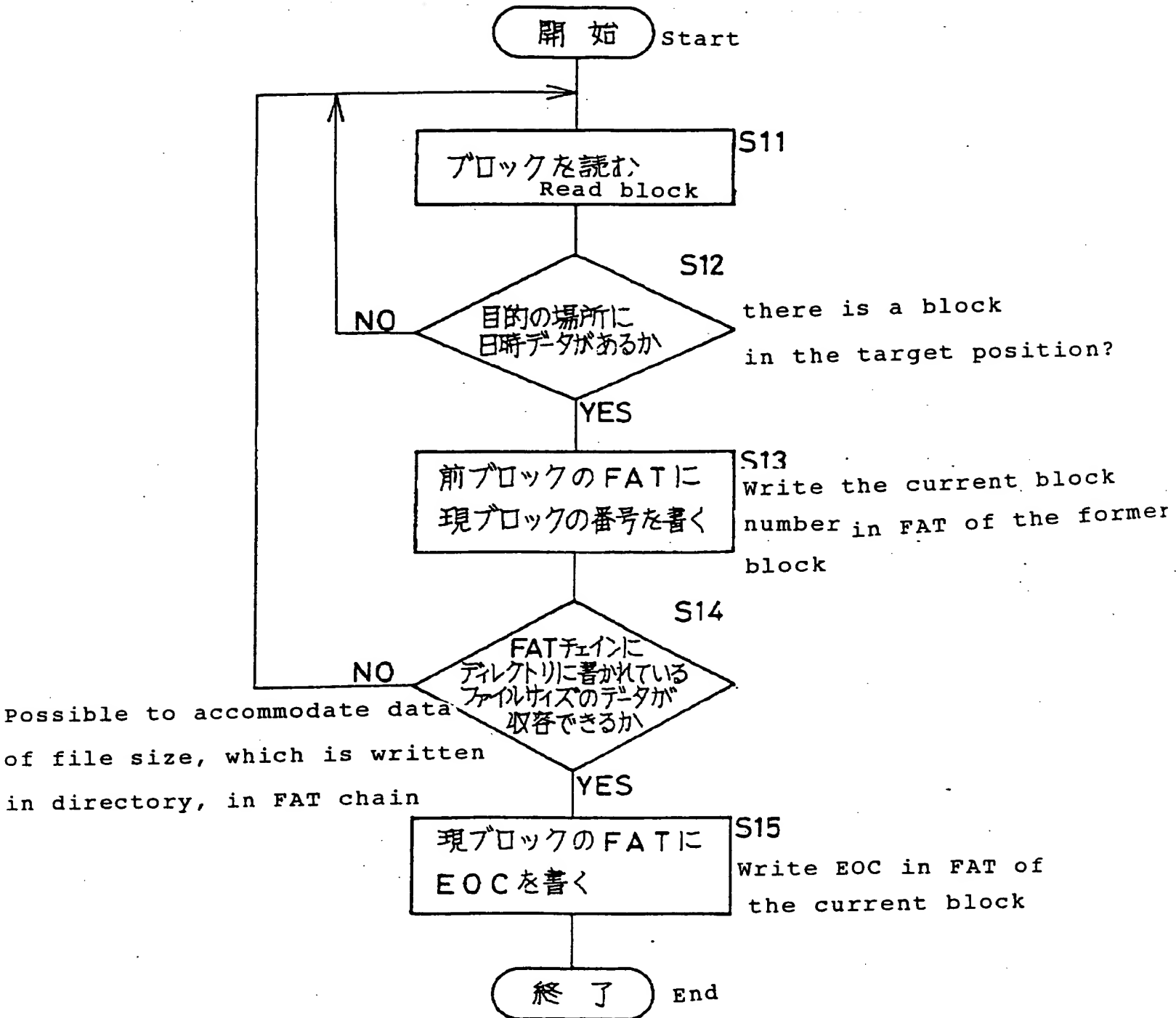
106 periodic execution control means

107 console

108 file access control means

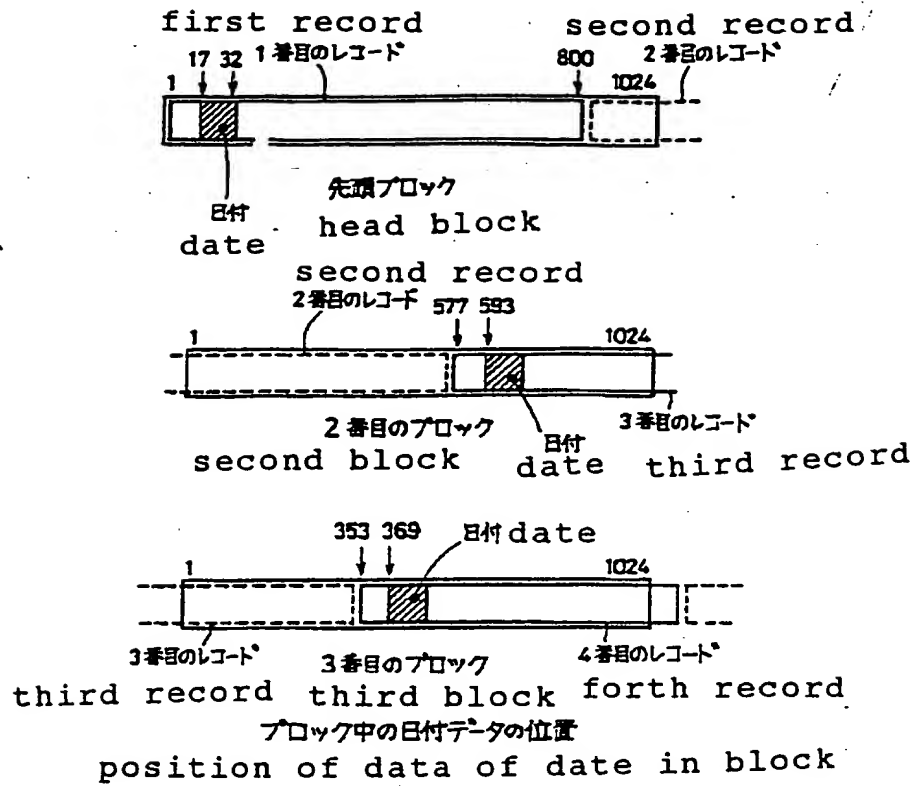
THIS PAGE BLANK (USPTO)

【図3】[Figure 3]



THIS PAGE BLANK (USPTO)

【図5】 [Figure 5]



【図18】 [Figure 18]

block number

ブロック番号 25 26 27 28 29 50 51

FAT 26 27 28 50 () 51 EOC

unused

データ
data

未使用 未使用

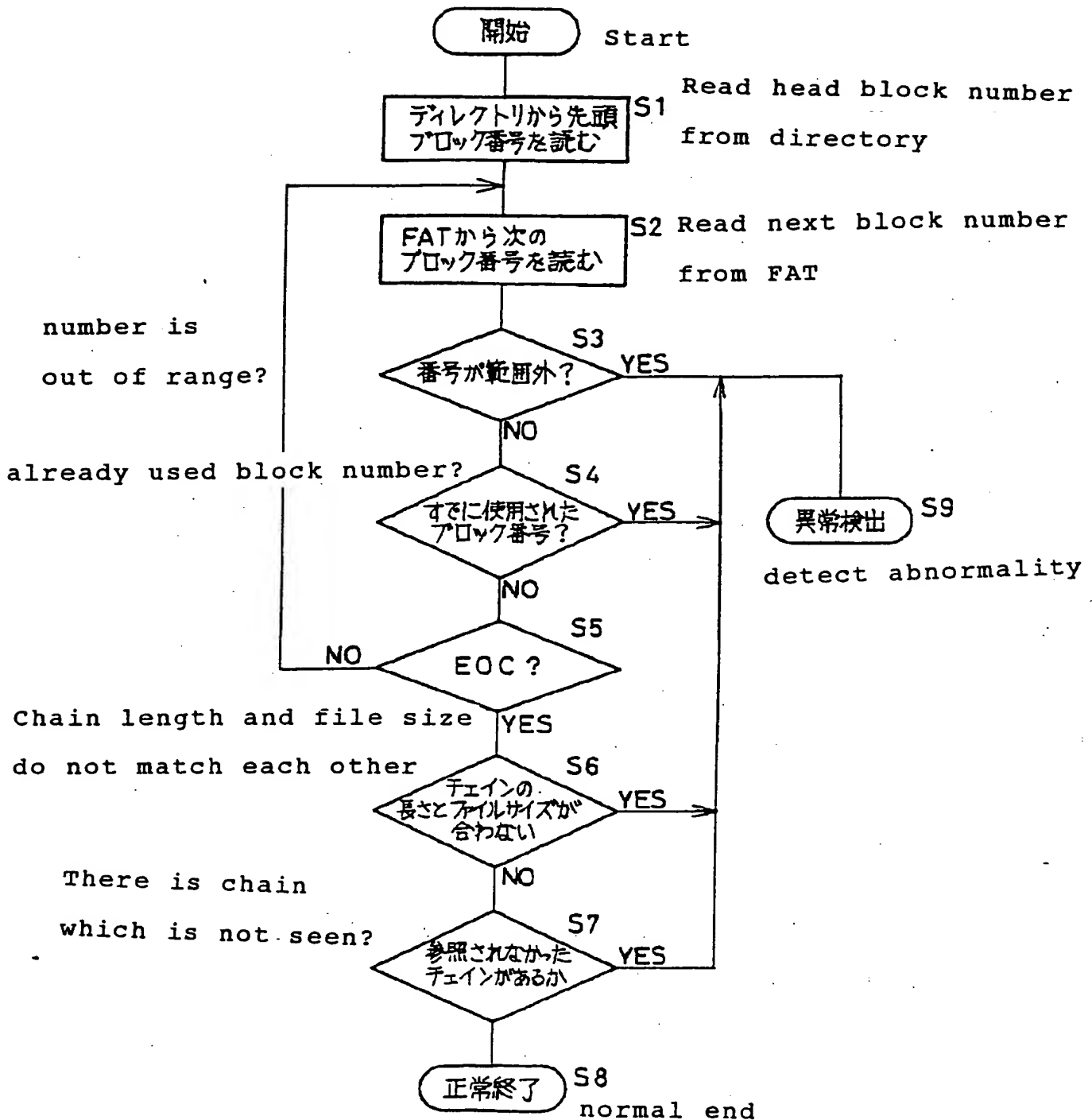
unused

FATとデータ

FAT and data

THIS PAGE BLANK (USPTO)

【図20】 [Figure 20]



ファイルシステム検査
file system check

THIS PAGE BLANK (USPTO)